Of Lions and Yakshis

Ontology-based Narrative Structure Modelling for Culturally Diverse Folktales

Franziska A. Pannach [a,*], Caroline Sporleder [a], Wolfgang May [a] and Aravind Krishnan [b]
[a] *Institute of Computer Science, University of Göttingen, Germany*
E-mails: franziska.pannach@stud.uni-goettingen.de, caroline.sporleder@cs.uni-goettingen.de, may@informatik.uni-goettingen.de
[b] *Department of Electronics and Communication Engineering, College of Engineering, Trivandrum, Kerala, India*
E-mail: aravindhp1999@gmail.com

Abstract. Vladimir Propp's theory *Morphology of the Folktale* identifies 31 invariant functions, subfunctions, and seven classes of folktale characters to describe the narrative structure of the Russian magic tale. Since it was first published in 1928, Propp's approach has been used on various folktales of different cultural backgrounds. We built an ontology that models Propp's theory by implementing narrative functions as classes and relations. A special focus lies on the restrictions Propp defined regarding which *Dramatis Personae* fulfill a certain function. We investigated how an ontology can assist traditional humanities research in examining how well Propp's theory fits for folktales outside of the Russian-European folktale culture. For this purpose, a light-weight query system was implemented using an Apache Jena Fuseki backend. In order to allow ontology browsing, we provide an institutional Webprotégé instance. To determine how well both the annotation schema and the query system works, we annotated twenty African tales, and fifteen tales from India. We evaluate the system by examining two case studies regarding the representation of characters and the use of Proppian functions in African and Indian tales. Our findings are in line with traditional analogous humanities research. This project shows how carefully modelled ontologies can represent and re-evaluate traditional theories of literary scholars, and how they can be utilized as a knowledge-base for comparative folklore research.

Keywords: Computational Folkloristics, Ontologies, Folktales, Narrative Structure, Vladimir Propp, India, Sub-Saharan Africa

1. Introduction

Folk and Fairy tales are a substantial part of oral folklore and an intangible element of cultural heritage.

They play an important role in the cultural heritage of regions, nations or cultural minorities. In a European context, fairy tales have been collected and edited by the Grimm brothers in the beginning of the 19th century [1] while their Russian counterpart Alexander Afanasyev collected more than 450 folktales of Russian and Slavic origin [2]. Afanasyev's tale collection later became the foundation of Vladimir Propp's theory *Morphology of the Folktale*, which was published in 1928, but only gained international momentum after being translated into English in 1958 [3].

This project aims to construct an ontology and a lightweight query system[1] for multi-cultural folktales following the morphological approach of Russian folklorist Vladimir Propp. In particular, we want to investigate:

1. The Proppian functions and character classes.
2. Specific folktale motifs from the Thompson-Motif-Index [4] (TMI) and tale type classes according to the Aarne-Thompson-Uther Index (ATU) [5].
3. The representation of the functions and motifs in selected African and Indian Folktales.

We want to investigate how a carefully modelled ontology can help assessing the differences between intercultural folktales with regard to their narrative

*Corresponding author. E-mail: franziska.pannach@stud.uni-goettingen.de.

[1] Our ontology and a light-weight query system is available at https://teaching.gcdh.de/ontology.

structure. For this purpose, a small corpus of African and Indian tales were annotated according to their Proppian functions and character representations. To achieve this, a system has been implemented that permits storing and querying tales with their metadata and with their annotations. An annotation contains the verbalisation of a Proppian function, e.g. the function β *Absentation* might be verbalised as "Each morning, Sibanda would sneak off to his food tree [...]." [6].

The ontology is open for additions of tales from these or other cultural backgrounds. Its expressive potential grows with the number of annotations. The ontology can assist different approaches on intercultural folktale comparison, e.g. how a certain Proppian function is verbalised in different tales, or how the verbalisation of a function in the same tale changes with translation. To this end, we annotated 20 mostly sub-Saharan African tales in their English translation, and 15 Indian tales both in Malayalam and English. We use this small corpus to illustrate the application of the system for comparative analysis.

A description of the project with a focus on African tales can be found in [7].

1.1. Outline of this paper

The paper ist structured as follows: the subsequent section introduces the application domain, i.e., Propp's *Morphology of the folktale* and a short introduction to the ATU and TMI indices. In Section 3, we discuss related work. Section 4 describes the used sources of folktales. Section 5 presents the design approaches of the query system, giving more insights in the implementation in Section 6. Section 7 discusses the information extraction from tale texts, as a proof-of-concept. Section 8 presents some results and describes some use cases from the application point of view. Section 10 summarizes the conclusion.

2. Description of the Domain

2.1. Morphology of the Folktale

The Russian folklorist Vladimir Propp introduced 31 invariant functions describing the morphology of the Russian magic folktale.

In his groundbreaking 1928 work *Morphology of the Folktale*, he introduces seven classes of Dramatis Personae, i.e. agents, within a story: *the hero, the donor*, who provides the *hero* with means to overcome the *villain*, *the dispatcher, the helper, the false hero, and the princess/her father*.

He argues that the narrative of folktales always follows the same pattern. Narrative functions, such as *XXXI Wedding W*, are strictly defined and specify recurrent units from which the tales are constructed. They follow a theory-inherent order, indicated by roman numerals. Furthermore they are identified by a literal or abbreviation which represents the function, e.g. *W* (*Wedding*), $\uparrow$ (*Departure*) or $\downarrow$ (*Return*). A specific function is always tied to specific Proppian characters, e.g. the Wedding function only applies if the *hero* character marries the princess (or a character that fulfills the narrative role of a princess). If a wedding between two other characters takes place, or if it appears at any other point than the end of the plot, the function does not apply.

A sequence of functions represents the plot of a tale and is encoded in a string of function literals, as shown in Sequences 1 and 2 in Section 2.2. According to Propp, a tale can consist of one or multiple sequences, in the latter case, he calls them *moves*. Moves can appear in chronological order, e.g. two subsequent stories within one tale, or a move can be interrupted by another move, e.g. in the form of a side plot.

Propp set four axioms [3, p.21-23]:

1. "Functions of characters serve as stable, constant elements in a tale, independent of how and by whom they are fulfilled. They constitute the fundamental components of a tale.
2. The number of functions known to the fairy tale is limited.
3. The sequence of functions is always identical.
4. All fairy tales are of one type in regard to their structure."

Furthermore, he grouped his functions into five categories: *Preparation*, i.e. the intial functions and first appearances of the main characters, *Complication*, in which the act of misfortune or villainy takes place, *Functions of the Donor*, where a helpful figur provides the hero with means to overcome the villain, the *Struggle* between Hero and Villain, and *Dénouement* in which the heros are rewarded for their action. Functions belonging into the preparation category are represented by Greek letters. A sequence of functions represents the plot of a tale and is encoded in a string of function literals.

We investigate those categories and specific functions when we discuss the results in Section 8 at the end of this paper, and exemplarily in the subsequent section.

2.2. Propp and African Folktales

Structural analysis of African folktales and the applicability of Proppian functions to them has not been introduced without critical side-eyeing. Daniel J. Crowley was sceptical whether Propp's approach was applicable to folklore studies, because in his opinion it was "doing too much violence to the variant nature of tales" [8, p.130].

Since then, studies that investigate the fitness of Propp's theory for African Tales have been conducted. A prominent example is the "Morphology of the Igbo Folktale" by Chukwuma Azuonye [9].

Azuonye published a morphological analysis of the *Obaraedo* tale in 1990 [9]. The same tale has been analysed by Ikechukwu Okodo in 2012 [10]. Both analyses explain how Propp's functions are represented in the *Obaraedo* tale. While Azuonye did not provide the full text of the tale, Okodo included the text translated to English in his article. From the explanations Azuonye gave on how the Proppian functions appear in the untranslated text, it becomes apparent that both of them worked with the same tale.

However, Azuonye's findings regarding the structure of the tale are significantly different from Okodo's analysis. Due to Propp's formalistic approach, we can easily compare both findings. Azuonye defines the function sequence of the *Obaraedo* tale as

$$\alpha\beta\gamma\delta\theta aBC \uparrow HIK \downarrow T. \tag{1}$$

While Okodo defines the sequence as

$$\beta\gamma\delta\epsilon\eta\theta aBC \uparrow FGHIK \downarrow \tag{2}$$

The first difference that we encounter when we compare both analyses is that the *initial situation* α does not appear in Sequence 2. The initial situation is often omitted since according to Propp it should not be regarded as an own function, but as "an important morphological element", which rather introduces the hero and the circumstances in which the tale takes place [3, p.25].

In Sequence 2, the preparatory functions include the tuple *Reconnaissance* ϵ, and *Trickery* η while in Sequence 1 *Violation* δ is directly followed by *Complicity* θ.

Following the *Departure* function, Okodo identifies the function *Provision or Receipt of Magical Agent F*. Propp's description of the tale defines this function as "The hero acquires the use of a magical agent." [3, p.43] Therefore, we can assume that Okodo sees the herbalist as the hero of the tale. In that case, Okodo's analysis of the tale is inconsistent, since he uses the *Departure* function when Obaraedo's father leaves the village to summon the herbalist. The *Departure* function, however, is designed for the departure of the *hero*. Both Azuonye and Okodo use the functions *Struggle H* and *Victory I*, when the herbalist/dibia, fights the spirit, indicating again that he fulfills the role of the hero in the tale. Additionally, they both define the *Departure* function as departure of the girl's father, but the *Return* function as a function of the herbalist/dibia.

In that sense, they are both separating the action described in the functions from the Dramatis Personae who fulfill them. This shows a rather free interpretation of what Propp clearly defined as "The Functions of the Dramatis Personae." [3]

These differences between two folkloric analyses show that the interpretation of Propp's functions is not universal, nor is there only one correct sequence of functions per tale.

2.3. Usage and Modeling Aspects of Propp's Morphology

Comparing analyses like those discussed above is tedious work and requires a lot of insight in the Proppian approach. As of yet and to our knowledge, there is no online system that allows to compare Proppian analyses of folktales to those of other tales or variants of the same tale. Therefore, contextualisation of Proppian analyses remains a manual task.

Our system permits storing the metadata about folktales and their publications, e.g. authors and editors, sources, or publishers, as well as the annotations according to Propp's morphology, according to a well-designed ontology. With this, it allows users to query Proppian functions and their verbalisations, and compare tales that share a function. Hence, we are providing folklorists with the neccessary information to form an opinion which of two or more divergent analyses analyses is more suitable for the given tale.

For this purpose, common vocabulary like SKOS [11] or the Dublin Core metadata schema [12] are used. Also, we use RDFS annotation properties like rdfs:label, rdfs:comment, e.g. to explain why a specific Proppian function was chosen.

Please note that we distinguish between *annotations* of the narrative patterns of the folktale in our domain, and the technical concept of *OWL AnnotationProperties*.

Furthermore, for the specific case of the comparison of the *Obaraedo* tale as described above, the author's comments on why a function was chosen were added to the function instance as an rdfs:comment. This way, users can comprehend the authors' reasoning on why a particular function was chosen.

Our ontology can facilitate a comparison between existing analyses like those by Azuonye [9] and Okodo [10], and those of tales from other regions. Existing annotations can be accessed easily through SPARQL queries or by accessing the triple search of the lightweight query system. Therefore, the system allows the study of Proppian morphology interculturally and language-independently which might lead to new findings in folktale research.

2.4. Motif Indices

The Aarne-Thompson-Uther index (ATU) [5] is used to classify a tale into exactly one class, the *tale type*. For instance, the tale of the *Frog Prince or Iron Henry* [1] falls into the ATU class *440 - The Frog King*, which belongs into the broader category *Magic Tales*. Type classes are relatively wide, describing the main story line of the tale. Therefore, each tale can only have one ATU type. Tale types also indicate the relation of tales that belong to the same class. The Aarne-Thompson-Index was first published in 1910, revised by Hansjörg Uther and republished as ATU-Index in 2004.

In contrast, the Thompson-Motif-Index [4] is more fine-grained, describing single motifs, i.e. "recognizable object[s], character[s], or event[s]" [8, p.127], such as characters, actions, or numerical patterns. A tale can contain more than one TMI motif, e.g. the *Frog Prince* tale includes motifs such as *B211.7.1 Speaking Frog*, *P40 Princesses*, *P23 Children and Parents*, *P320 Hospitality*, or *D935 Transformation: Frog to Person*.

3. Related Work

Peinado et. al [13] modelled a description logic ontology based on Proppian functions. Their work is probably the closest to our project. They implemented Proppian functions and some additional subconcepts regarding persons, places and objects as ontology classes. Their description logic foundation was used to generate folktale plots. Similar to our project, a significant amount of additional domain knowledge was modelled to achieve their goal of plot generation.

In order to achieve a temporally sound story line, they used relations and concepts from the CBROnto case representation structure. [14] In a two stage approach, they first generated a raw plot from Proppian functions which were then filled with a textual representation.

Since our goal was to use our ontology for narrative annotation of folktales and queries on those annotations, we modelled our ontology slightly different. They made some design choices we could not reuse for our approach, e.g. the ontology class *Move* is a subclass of *ProppFunction*. In our case, *Move* is a subclass of *Tale*, because a tale consists of one or multiple moves, i.e. story lines. A *Tale* or *Move* has a function sequence, which we modelled as a Data Property.

Declerck et. al [15] created an ontology that modelled Proppian functions as classes with a vast set of rdfs labels in English, German and Russian, ProppOnto. The Internationalized Resource Identifier (IRI) of their classes is a short description of the function according to the corresponding literal, e.g. *Delta1*. The functions in Declerck's work are not grouped into the five categories defined by Propp. Furthermore, they did not provide object properties for functions, such as those modelling sequential order or those that connect a function with the corresponding tale it appears in. The extensive labels and rdf:comments provided by [15] were found a very useful addition to our existing ontology. Therefore, they were added to our ProppOntology and converted to skos:prefLabels for the English labels, and skos:altLabels for the German and Russian labels.

In addition to the ProppOnto, Declerck et al. [16] built an ontology on the Thompson-Motif-Index in combination with Aarne-Thompson-Uther Types. These two indices are core instruments for traditional folktale research, as described in Section 2.4.

Declerck et. al provide rdfs:labels in English and German, motifs and types appear both as classes in the ontology and as individuals. [17] Their ontology also includes a set of additional motifs as defined by the ETrap project[2]. As for now, only the core ATU classes and TMI motifs have been imported into the ProppOntology. While the population of the ontology with motifs and tale types as individuals technically does not follow the approach in our case, they were nevertheless

[2] https://www.etrap.eu/

imported to allow possible reuse at a later stage in the project. Following the approach of our project, motifs and tale classes should have only been added as individuals if they occur in a tale that is annotated, with a distinction in naming, such as *O_ 2012_F* for the appearance of the evil spirit in the 2012 version of the *Obaradeo* tale. This specific individual can then hold further information on how it is verbalised in the given tale.

Nikolina Koleva [18] built a folktale ontology that modelled family relations of characters. She used SWRL[3] rules for the classes to allow ontology reasoning. These rules were then used to automatically populate the ontology by iterating through a text, searching for semantic cues that introduce fairy tale characters. Koleva used NooJ grammars to detect the entities and the OWL API to populate the ontology. The automatic extraction of characters and role attribution from fairy tale texts seemed to work comparatively well, albeit on a small number of folktales.

However, her characters lack certain features such as their verbalisation, or information about the tale they appear in. Therefore, in this project, we tried to implement the semi-automatic population in a different manner. Nonetheless, the class hierarchy of the characters was imported to the ProppOntology as it is a valuable addition to the ontology beyond Propp's definition of character roles, especially with regard to the repeated motif of family relations in folktales from different origins.

Additionally, Koleva provided labels, although mistakingly annotated as dc:language fields, for the character classes in German, English, Russian and Bulgarian. Those fields were carefully transformed into skos:prefLabel and skos:altLabel.

Even though the family ontology provided object properties *hasWife* and *hasHusband*, the classes modelling spousal relationship were missing, and had to be added to the ontology.

Furthermore, her approach was focussed on a specific tale (*The Magic Swan Geese*), which resulted in her making assumptions regarding the extraction rules that do not necessarily hold true for other tales, e.g. that a mother is a woman which is not necessarily the case when we think about fabulas or animal tales.

[3] Semantic Web Rule Language, https://www.w3.org/Submission/SWRL/

4. Sources

The African tales were taken from a number of anthologies. We tried to find a "healthy" mix between scholarly collections of tales, and typical children's stories. Therefore, we included Harold Scheub's collection *African Tales* [19] (2005), Nick Greaves' children's book *When the Hippo was hairy and other tales from Africa* [20], Children of Wax (1989) [6] by Alexander McCall Smith, and Phillis Savory's *Bantu Folk Tales From Southern Africa* (1974) [21].

Secondly, we identified a small corpus of Indian tales, from the state of Kerala, published in Malayalam with their English translation. The tales in Malayalam have predominantly been taken from *Aithihyamaala* [22], a corpus of all the prevalant legends in Kerala written in the 20th century. All the stories, history, mythology, and romance of the Keralite community of the time, are presented in 126 articles. It represents the social and cultural life in the state at that time, and popularised characters like *Kayamkulam Kochunni*, *Naranathu Bhrandan* and *Kadamattathu Kathanar*.

The book is still an indispensable reference for historians of the Keralite society, which lacks in historical record keeping. The English versions of the tales have been extracted from a translation of the book *Aithihyamaala, The Great Legends of Kerala*. [23]

To encompass poetical literature in the scope of the study, some stories have been taken from the famous *Vadakkan Pattukal*, a collection of Ballads in Malayalam. These have survived by oral passage from generation to generation, and are believed to have been written down in the 17th or 18th century. There may have been some additions or reductions over time, but they still remain largely intact. The epic poem *Poothapattu* has also been included in the corpus. [24] [25].

5. Design

The choice to model Propp's theory by using an ontology has two main motivations. Firstly, the functions are highly hierarchical as they are divided in categories, functions and subfunctions. Secondly, the use of an RDF model allows us to represent the Proppian functions not only as classes within the ontology, but to additionally model them as object properties between two instances of subclasses of *Dramatis Personae*. Therefore, each Proppian function has two representations within the ontology. As an example, the

hero Kathanar kills the Yakshi in the Indian tale *Kathanar and the Yakshi*. We use the object property *defeats* to connect the two instances of *Hero* (Kathanar) and *Villain* (Yakshi). Additionally, we annotate the function *Victory I* as an individual that holds the verbalisations in both English and Malayalam as shown in Fig. 1.

This approach allows us to query not only instances of functions, but also the relation they represent between characters in a tale. After all, the functions are defined as "Functions of the Dramatis Personae" [3] and should therefore not be separated from the characters in a tale. To our knowledge, the representation of functions as object properties as followed in this project is a novel approach.

To demonstrate how a thoroughly modelled ontology in combination with natural language processing approaches can be employed to semi-automatically populate the ontology, an information extration component for folktale characters and Proppian functions has been added. This module should be seen as a proof of concept study rather than a perfect tool for extracting information from folktale texts. The implementation of the ontology-guided information extraction is currently not accessible on the project website.

Instead of using the information extraction tool, manual annotation of folktale texts can be used to populate the ontology with additional folktales.

5.1. Competency Questions

For the design of the ontology, we followed Noy and MacGuinness' approach and defined a set of competency questions [26]. If these questions can be answered by the final ontology, it has fulfilled its expressive purpose. They should be seen as a minimal requirement to the expressitivity of the system.

1. Which folktales fall into a given motif class, e.g. ATU 70-99 Other Wild Animals?
2. Which Dramatis Personae appear in a given tale?
3. Which Proppian functions appear in African folktales?
4. How are Dramatis Personae interacting in the African folktales, e.g. which figures use the "interdiction" relation?
5. Which sequences of Proppian functions appear in a given tale? Which sequences appear in tales in general?
6. Which Proppian functions follow a given function predominantly, i.e. are there patterns withing the Proppian sequences?
7. Who is the editor of an anthology with folktales from a given origin?
8. How are Proppian functions verbalised, i.e. which words are used to describe events that fall into a given function class?
9. Is there a dominating interaction between certain classes of Dramatis Personae?

5.2. Axioms

Following Noy and MacGuiness design pipeline [26] further, a set of axioms was defined before the implementation of the ontology. We determine some axioms regarding the publication of the tale and its metadata:

- Each tale is published in an anthology, or as part of a journal article.
- Each anthology has at least one editor, a title, a publisher, and a date of publication.
- Each tale has a title.
- A tale can have an author and an origin if known.
- Each tale falls into one of the ATU type classes.
- Each ATU class has an ATU number and a description.

Furthermore, we defined content-related axioms:

- Each tale has a set of Dramatis Personae.
- Each fictional character belongs to one or more character classes and is represented by one or more verbalisations.[4]
- If a Proppian function applies to a tale, there is some verbalisation in the text.
- Proppian functions follow a specific order (see below), this order is represented by a sequence.
- Each Proppian function is represented by a symbol.

In addition to these axioms, following Propp's approach, we derive axioms for the description of the narrative. These restrictions mainly model the scope of Proppian functions, e.g. the *Wedding* function can only be applied if it describes a relation between the *Hero* and the *Princess*. If a function applies to a tale, the axiom holds. Not all of the functions/axioms need to be fulfilled by every tale. However, their order needs to remain the same.

[4] e.g. in the tale Snow White 'the stepmother' and 'the evil queen' describe the same individual

Figure 1. Webprotégé View of the annotation of the function *Victory I* for the tale *Kathanar and the Yakshi* and its verbalisations

5.3. Modelling Folktales in Description Logic

Modelling folktale narrative in Description Logic was particularly challenging, since certain real-life restrictions do not necessarily hold for the folktale domain. For instance, while in real life the classes of humans and animals would certainly be distinct, these classes might mix in folktales. Transfiguration of humans into animals or a human mother giving birth to animals are recurrent pattern especially in African tales.

Especially with regard to future extensions of the ontology, we needed to make sure that the logical foundations are not preventing the annotation of unforeseen patterns in folktales. Therefore, only general description logic statements, such as those that are indicated by Propp's theory, have been defined in awareness that this approach might lead to a limited application of ontology reasoning in the future.

First, we defined a set of description logic statements that model the class hierarchy. We distinguish between statements that are content-related, such as *Princess* $\sqsubseteq$ *DramatisPersonae*, and those that are metadata related, such as Anthology $\sqsubseteq$ Publication. Secondly, since our aim was to model Propp's functions not only as classes but also as relations between folktale characters, we defined restrictions regarding the range and domain of Proppian functions, e.g. Donor $\sqsubseteq \forall$ requestsService.Hero.

Table 1 shows how we modelled Proppian functions as relations between characters. In addition, we modelled functions and their subfunctions as ontology classes. Since the function hierarchy follows directly from Propp's theory [3], we refrain from listing description logic statements on the class hierarchy.

5.4. Implementation of Ontology Classes

Fig. 2a shows how the 31 function classes were implemented. They are divided into the five main categories *Preparation*, *Complication*, *Functions of the Donor*, *Struggle*, and *Dénouement*. Fig. 2b shows the main character classes, in particular the Proppian characters and the classes imported from [18]. The subclasses of *Animal* are far from complete, and are extended where needed.

In contrast to ProppOnto by [16], the classes modelling the Proppian functions and their subfunctions have been named after their original description as published in [3].

Furthermore, in our case the types of Dramatis Personae are modelled as subclasses and not as individu-

Table 1
Important Concepts of Proppian Functions

Concept
FictionalCharacter ⊑ ∃ appearsIn.Tale
∃ FollowedBy.ProppianFunction ⊑ ProppianFunction
∃ PrecededBy.ProppianFunction ⊑ ProppianFunction
∃ correspondsTo.ProppianFunction ⊑ ProppianFunction
ProppianFunction ⊑ ∃ applies.Tale
FictionalCharacter ⊑ ¬ RealPerson
Hero ⊑ ∀ acquires.MagicalAgent
Hero ⊑ ∀ acquiresFrom.Donor
Hero ⊑ ∀ attackedBy.Donor
Donor ⊑ ∀ attemptsHarm.Hero
Donor ⊑ ∀ begsForFreedom.Hero
Villain ⊑ ∀ causesHarm.FamilyMember
False Hero ⊑ ¬ Hero
FamilyMember ⊑ ∀ relatedTo.Hero
Villain ⊑ ¬ (Victim ⊔ Hero)
Hero ⊑ ∀ brandedBy.Object
Hero ⊑ ∀ combats.(Villain ⊔ Donor)
Reward ⊑ ∃ isRewardedTo.Hero
Reward ⊑ ¬ MagicalAgent
Villain ⊑ ∀ deceives.Victim
Princess ⊑ ¬ Princess'Father
Hero ⊑ ∀ defeats.Villain
Hero ⊑ ∀ devidesProperty.Donor
Hero ⊑ ∀ frees.Victim
Villain ⊑ ∀ gainsInformation.Victim
FictionalCharacter ⊑ ∀ interdicts.FictionalCharacter
Hero ⊑ ∀ interrogated.Donor
Hero ⊑ ∀ marries.Princess
Hero ⊑ ∀ offered.MagicalAgent
MagicalAgent ⊑ ∀ offeredBy.Donor
Villain ⊑ ∀ persuades.Victim
Task ⊑ ∃ proposedBy.FictionalCharacter
Task ⊑ ∃ proposedTo.Hero
Hero ⊑ ∀ pursuedBy.FictionalCharacter
Hero ⊑ ∀ reactsTo.Donor
Hero ⊑ ∀ recognizedByMeansOf.Object
Villain ⊑ ∀ reconnaissance.(Victim⊔ Object)
Donor ⊑ ∀ requests.Task
Donor ⊑ ∀ requests.Object
Donor ⊑ ∀ requestsFrom.Hero
Donor ⊑ ∀ requestsMercy.Hero
Donor ⊑ ∀ requestsService.Hero
Hero ⊑ ∀ rescuedBy.FictionalCharacter
Hero ⊑ ∀ resolves.Task
Villain ⊑ ∀ taskesPosessionFrom.Victim
Hero ⊑ ∀ testedBy.Donor
Villain ⊑ ∀ threatens.Victim

als of the Dramatis Personae class. This way, we can assign characters appearing in a specific tale as individuals of character classes, such as *O_Obaradeo* as *Victim*.

Following Propp's naming conventions, the subfunctions are named following the same pattern as the parent function, e.g. $\delta 1$*_Interdiction_violated*. The ontology was designed in OWL format using the Protégé desktop application [27].

Alternative labels consist of translations of the skos:prefLabels in different languages, such as German, Russian, and Bulgarian, that were either imported from the Family Ontology [18] or the ProppOnto[16], provided by native speakers of isiZulu for the possible application of the system for African tales in their native languages, or created ourselves. Some English skos:altLabels have been derived from WordNet synsets via the NLTK WordNet interface[5], in order to increase number of matches between the folktale text and the skos:prefLabels for the information extraction.

Example specifications for function and character classes are given in Listings 1 and 2.

Some classes appear in pairs, such as the *A Lack* function and *K Liquidation of Lack*. They can be combined using the *correspondsTo* relation.

5.5. Object and Data Properties and their Constraints

As mentioned before, we modelled Proppian functions as classes, capturing their appearance in a tale as individuals. To be able to examine the interaction between folktale characters, we also modelled them as object properties.

Propp [3] defines strictly which character has to perform a certain action in order for a function to apply. For instance, the *Hero* can be only interrogated by the donor, which implies the function *D2 Donor greets and interrogates the Hero*. If another person e.g. the villain interrogates the hero, in order to find out more about him or her, the function *Reconnaissance* $\epsilon 1$ applies.

Data properties mainly provide metadata information, such as the tale title or the key used for distinguishing the invididuals. A few data properties come with the Family Ontology, such as *hasGender*. We did not strictly annotate those features for the African tales, since they have been included before the Fam-

[5] http://www.nltk.org/howto/wordnet.html

Listing 1: Example specification of a Proppian function

```
###  https://teaching.gcdh.de/ProppOntology/1.0.1#XVIII_I_Victory
ProppOntology:XVIII_I_Victory rdf:type owl:Class ;
                              rdfs:subClassOf ProppOntology:IV_Struggle ;
                              rdfs:comment "Der Gegenspieler wird besiegt."@de ,
                                           "The villain is defeated."@en ,
                              skos:altLabel "Sieg"@de ,
                                            "ukunqoba"@zu ,
                              skos:prefLabel "Victory"@en ;
                              skos-xl:literalForm "I" .
```

Listing 2: Example specification of a Character class

```
###  https://teaching.gcdh.de/ProppOntology/1.0.1#Villain
ProppOntology:Villain rdf:type owl:Class ;
                      rdfs:subClassOf ProppOntology:Dramatis_Personae ;
                      dc:source "Vladimir Propp: Morphology of the Folktale. Austin, Texas 1968"@en ;
                      skos:altLabel "Schurke, Boesewicht"@de ;
                      skos:prefLabel "Villain"@en .
```

ily Ontology was imported. For the Indian tales, such as *Kathanar and the Yakshi*, we tried to include those information where available.

5.6. Folktale Annotation

Please note that the use of the term annotation in this section follows the linguistic definition, not the sense of "annotation properties".

All in all, we asked five different annotators to provide Proppian analyses for different folktales. Annotators were asked to determine Dramatis Personae and their respective Proppian roles, as well as Proppian functions as they appear in the tales. Each character and function instance was annotated as an individual in the ontology. They are identifiable by a key that indicates to which folktale they belong, e.g. individuals starting with *COW* belong to the tale *Children of Wax*. Despite a function instance always being connected to a tale by an *applies* relation, respectively a character by a *appearsIn* relation, this naming convention can be used for filtering query results later on and helps keeping the list of individuals comprehensible.

Each function or character of a tale comes with a verbalisation. In the case of annotations of the same tale in more than one language, we provide verbalisations in both languages, e.g. in English and Malayalam. We believe this feature allows interesting insights in the cultural transfer that folktales undergo during the translation process.

Furthermore, annotators were asked to provide metadata of the tale, such as the title or the publication it was published in.

6. Implementation

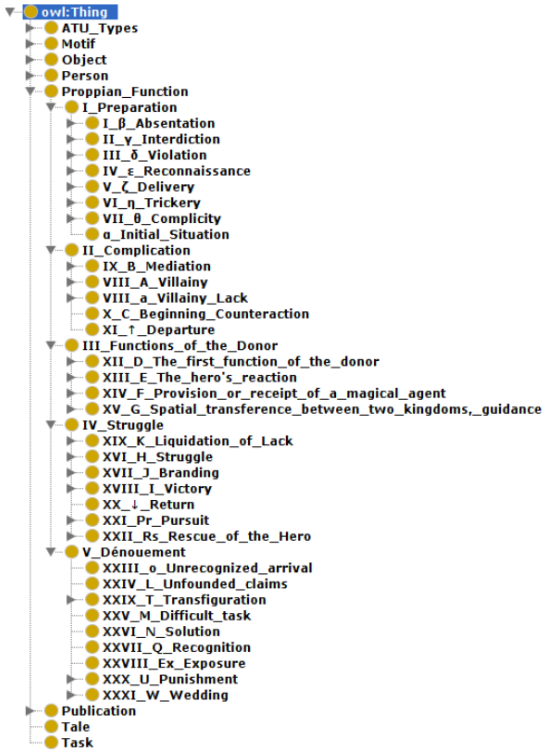
The activity diagram in Fig. 5 depicts the general functionality of the different system components.

The core of the system is a Flask[6]-based web application which provides three major functionalities: queries, annotation and ontology browsing. While most modern web applications are developed using programming languages like PHP or Ruby, Python was used in the context of this project because of the extensive availability of libraries and toolkits especially for the information extraction. This way, the system was developed in one language, avoiding the need to exchange data back and forth between different applications written in different programming languages.

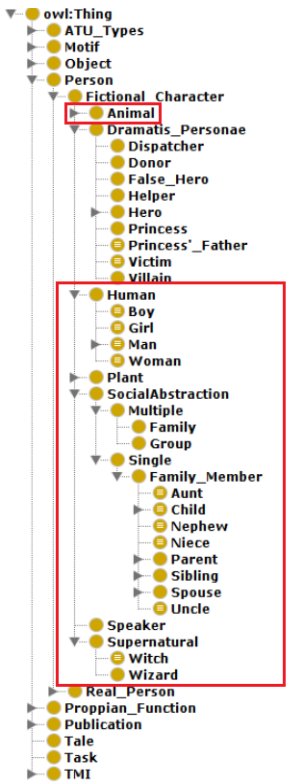
The Flask application builds the webpages from HTML templates, and communicates with the Fuseki webserver via a RESTful API. The Fuseki server processes the SPARQL queries and sends the results back to the Flask application.

For the ontology processing, an Apache Jena Fuseki server application is used. It provides comfortable handling of SPARQL updates and queries via a REST-

[6] http://flask.pocoo.org/



(a) Subclasses Proppian Functions



(b) Subclasses of the Dramatis Personae, red boxes indicate classes imported from the Family Ontology [18]

Figure 2. Subclasses Function and Dramatis Personae

ful API. For development purposes, the employment of Fuseki came with the advantage that its interface could be used to check if the ontology-driven information system that was developed behaves as desired, especially for the verification of the queries.

For the productive system, the Fuseki server is hosted on a port that is only accessible from the server on which the Flask application is deployed. That way, we ensure that no requests, especially no SPARQL updates, are sent to the RESTful API except those that come from the Flask application. This way, the risk of harmful injections into the ontology is reduced.

Webprotégé is used for the ontology browsing and annotation part of the system [28]. While querying in itself already provides a lot of insight, especially a good overview of individuals that were added, users might want to see how classes and subclasses are defined. The Webprotégé instance is not directly connected to the Flask application. A MongoDB database is used for managing Webprotégé user accounts.

6.1. SPARQL queries

The users of the ontology-driven information system have three means of querying the ontology. A basic text field can be used for advanced queries, triple queries can be used to investigate relations between rdf triples, and single queries provide means to investigate single classes.

Conveniently, Jinja 2 provides means to auto-escape user inputs in a way that prevents potentially harmful inputs. To reduce the risk further, the Flask module which delivers the query pages only allows GET requests to the Fuseki API.

The user can provide a syntactically correct SPARQL query, including prefixes, interpunctuation, query limits or regex restrictions in the text field.

All queries are processed through the Python package SPARQLWrapper2 which is used by the Flask application to send the query to Fusekis sparql endpoint (ds/sparql) as a GET request.

The second way of querying the ontology is provided by a simple graphical user interface. Users can fill a triple query pattern and enter either one or two classes, leaving the ones empty that would be represented by the placeholders in a SPARQL query as shown in Fig. 6. The first and third field are dedicated to classes, while the second field is assigned to the relation. When the query page is loaded, the most recent relations, ranges and domains are queried from the ontology to create a dropdown menu for each of

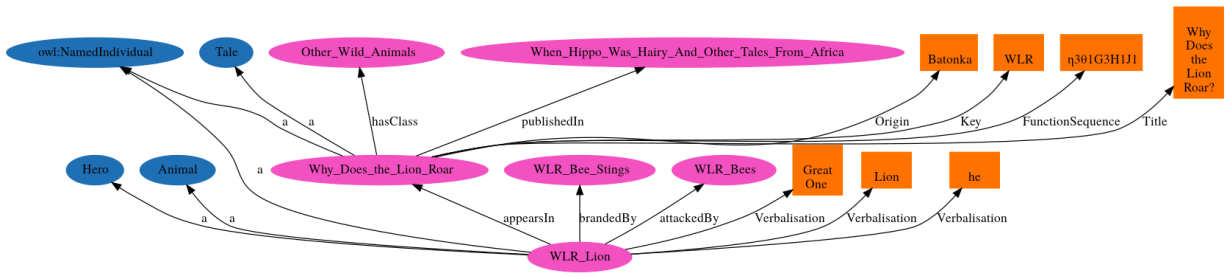


Figure 3. Example graph for tale and character (blue ellipses indicate classes, pink ellipses stand for individuals and data property values are indicated by orange boxes)

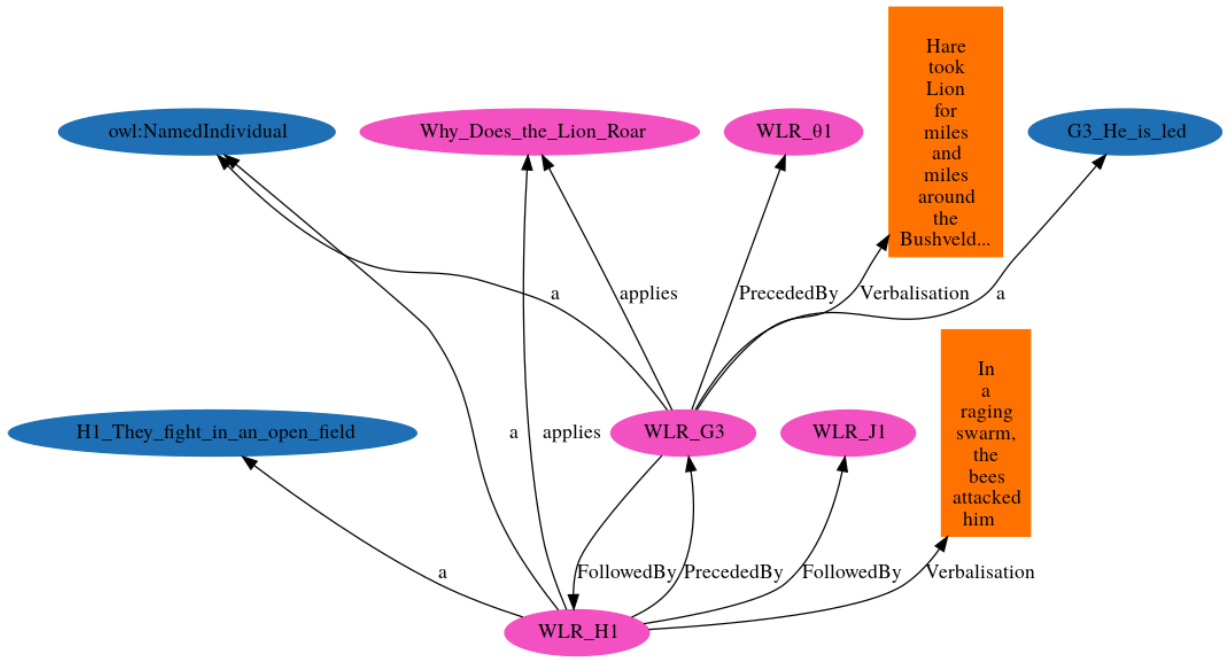


Figure 4. Example graph for Proppian functions

the fields. A star at the end of a class name is used as a flag to query not only the class itself but also its subclasses. If the checkbox behind the first or the last of the fields is ticked, the query looks for individuals instead of classes.

All query results from either of the three ways to query the ontology can be exported as a CSV file.

7. Information Extraction from Tale Texts

We attempted to extract some of the information encoded in the text semi-automatically. Specifically, nominal phrases that describe characters or animals, and instances of Proppian functions were of interest. On the other hand, nominal phrases of non-living objects that are repeated through the text can indicate a motif, such as the tree that *Cinderella* repeatedly visits which supplies her with the ball gown [1] corresponds to the TMI motif *D950 Magic Tree*. As of yet, we focussed on the extraction of characters and instances of Proppian functions and leave the motif extraction efforts for a future project.

While Wimalasuriya and Dou argued that linguistic extraction rules should be part of the ontology [29], we implemented the natural language processing elements entirely on the Flask side of the application. With a rule based approach, e.g. using regular expressions or gazetteer lists, it would make sense to include it within the ontology. However, this project followed a machine

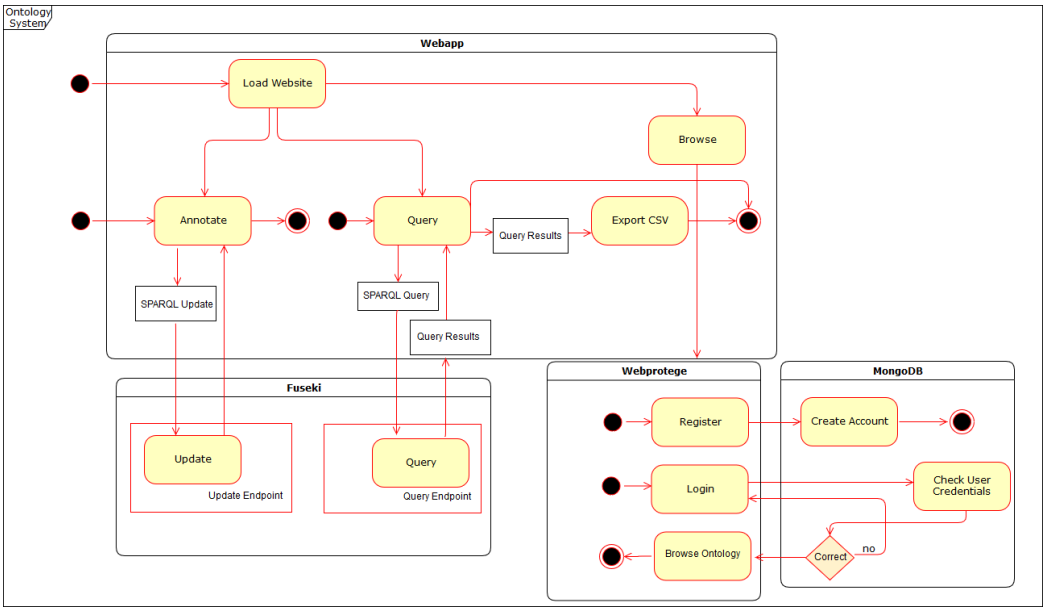


Figure 5. Activity Diagram of the Information System's functionality

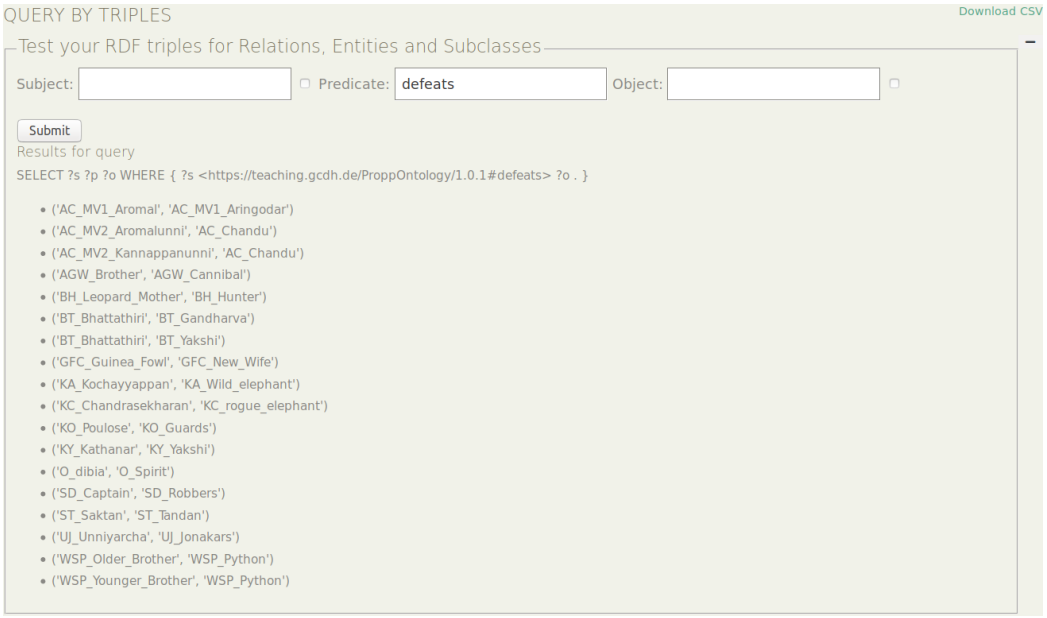


Figure 6. Example of the triple query functionality

learning approach that used the Python module NeuralCoref [7].

[7] https://huggingface.co/coref/

7.1. Entity Recognition for Folktale Characters

Initially, a set of semantic rules were defined to extract potential candidates of *Dramatis Personae* from the text. However, this approach did not yield satisfying results. The main reason might be that the rules for

the appearence of characters in tales must naturally be relatively broad.

A rule like: $NP : < DT >? < JJ > * < NN >$, would deliver correct nominal phrases, such as *the girl*, but also yield many false positives, since not every gramatically correct nominal phrase indicates a folktale character. Stricter, more sophisticated rules would likely not find entities that are verbalised in a simple manner, like *the man*.

The NLTK toolkit for Python provides a named entity chunker *ne_chunk*[8]. Expectedly, fairy tale texts do seldomly supply *named* entities, with exception of some popular tales like *Hans in Luck* or the *Obaradeo* tale discussed before. Usually, characters are introduced in a more general way, e.g. *the girl*. Therfore, the pure named entity recognition task was abandoned.

Since verbalisation of characters is one of the interesting features the ontology is supposed to supply, the focus shifted to the resolution of coreferences instead. The main idea behind using coreferences was that entities or other important features will likely be repeated throughout the text. We hypothesised that instances of *Dramatis Personae* yield particularly long coreference chains since they are key elements in folktale plots.

A satisfyingly working coreference resolution tool would not only provide characters that occur in the text, it would also provide reoccuring motifs, e.g. a tale revolving around an apple tree would yield many coreferences for *apple tree* or *tree*. Using a coreference approach yields results for named entities as well as unnamed entities, which is the most significant advantage and the main reason this approach was chosen.

From the available coreference resolution approaches, the NeuralCoref approach method was found to be the most promising. They trained their word embeddings model on the OntoNotes corpus. Making use of two neural networks, they calculate a score for pairing a word with an antecedent or not.

Although NeuralCoref was initially designed for coreference resolution in chatbot systems, this approach seems to work reasonably well on English folktale texts.

The text is first preprocessed using Spacy's *nlp* method[9]. Subsequently, coreferences are resolved using NeuralCoref. Candidate entities of characters are identified from the text using Spacy's named entity recognition method *ents*, finding token sequences from the entire text that are labelled as *Person*. Since this list alone yields very noisy results, the candidate tokens are then compared to the antecedents in the coreference clusters. Candidates that do not appear in the coreference clusters are abandoned. Certainly, this comes with losing some of the classes. However, only considering the tokens labelled as 'Person' yields too many false positives.

[8] https://www.nltk.org/api/nltk.chunk.html
[9] https://spacy.io/api/doc

7.2. Extraction of Instances of Proppian Functions

For extracting occurences of Proppian Functions, the extensive SKOS labels provided by the ontology were employed. For the time being, only skos:prefLabel fields are used. However, skos:altLabels could be used to identify instances for classes in different languages in the future.

For the information extraction, the text is preprocessed as described above. A SPARQL query yielding the values of all pref:Labels and their corresponding classes is sent to the Fuseki server at the beginning of the text processing.

After the coreferences are identified, a list of first mentions in all the coreference chains is created. Each mention is tokenized and stripped of punctuation. A list of tokenized prefLabels is created. Both lists are then lemmatized using the NLTK WordNetLemmatizer and compared. If one antecedent matches a token in a prefLabel, it is added to the list of potential candidates for that particular class.

The results of both approaches are then handed back to the Flask application, which creates an input form. If a potential person is found, a dropdown list allows the user to select the correct ontology class. If a candidate class is found by the second approach, the class name is shown next to the input field. Users can then change the data and create their own annotation.

8. Results and Evaluation

This section reports the results of the application of the light-weight ontology query system applied to African and Indian folktales. To date, the corpus of annotated tales includes 20 (mostly sub-Saharan) African tales and 15 tales from the region of Kerala in southern India.

We want to investigate the annotations with respect to the structure of the tales, paying particular attention to initial and final functions, and see how characters are represented in culturally different tales.

However, to ensure we do not fall into the trap of believing that we as non-folklorists "are qualified to speak authoritatively about folkloristic matters" [30, p.401], we focus on reporting the verifiable results, leaving deeper interpretations of our findings to the interested folkloristically educated scholar.

It should also be borne in mind that our corpora are relatively small and results can therefore only be indicative of potential tendencies that would have to be verified on a larger corpus.

We believe that the results the ontology system can yield are potentially interesting for folklorists who want to compare Proppian analyses of African and Indian tales. Furthermore, existing theories about those tales can be verified or falsified.

To evaluate the ontology, we are phrasing natural language questions as SPARQL queries to our lightweight query system.

For security reasons, we did not make the Fuseki server accessible for queries. Instead queries are made from our webinterface and internally forwarded to the server.

It has to be noted that our front end is not as robust against faulty SPARQL queries as the Fuseki frontend. Therefore, all queries have to be syntactically correct, especially the corresponding prefixes have to be given. We plan on extending the functionality of the frontend in the future, e.g. by adding visualizations for the data that is currently only displayed in a list.

8.1. Case Studies

8.1.1. Representation of Characters in African Tales

Characters in the annotated African tales mainly belong to three upper classes, *Animal* (23), *Family Member* (24) and *Dramatis Personae* (45). Of course, one character can belong to multiple of those upper classes. For the tales from India, the most prevalent character classes are *Human* (44) and *Dramatis Personae* (55). The dramatis personae fall into seven categories as defined by [3]. Fig. 7 shows the distribution of Proppian characters in the corpus.

The classes *Hero* and *Villain* appear 16 and 17 times in the corpus of twenty African tales. Five instances of *Victim*, three instances of *Donor*, and two instances of *Helper* occur; *Seeker*[10] and *Dispatcher* both appear exactly once. There was no instance of either *Princess* or *Princess' Father*.

[10]The role of the seeker is a specification of Hero and could therefore also be counted into the Hero class.

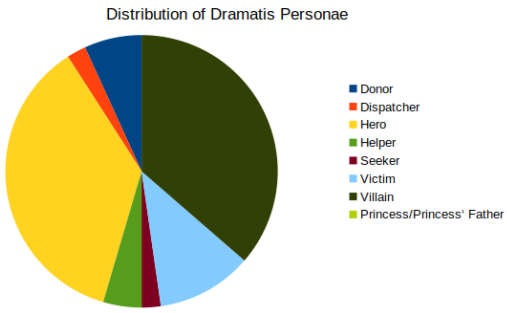


Figure 7. Distribution of Dramatis Personae in African tales

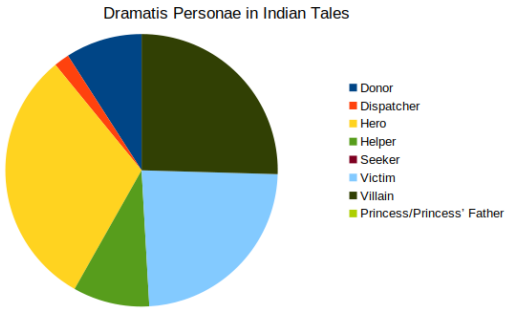


Figure 8. Distribution of Dramatis Personae in Indian tales

In our corpus of fifteen Indian tales, we find that the most common Proppian characters are *Hero* (17), *Villain* (14), and *Victim* (13). In addition, the classes *Donor* and *Helper* appear five times each, and there is one occurence of the *Dispatcher* class.

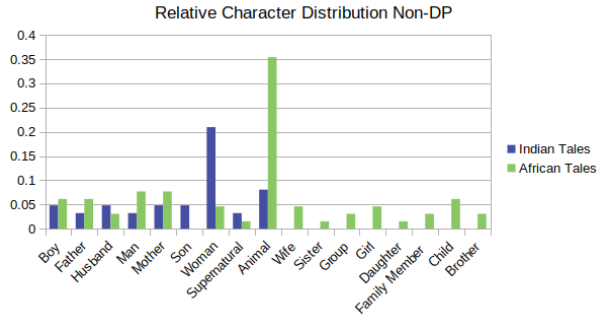


Figure 9. Non-Dramatis Personae Character Classes in African and Indian Tales

Since we imported the Family Ontology [18] to gain more insights on how Proppian characters are represented in tales, we allowed characters to belong to more than one character class. For instance, if the victim in a tale is the father of the hero, his character

might fall into the classes victim, man, and father, and husband if the hero's mother appears in the tale as well.

Fig. 9 shows character classes in both Indian and African tales that do not belong into the group of Proppian Dramatis Personae. While African tales show a preference for animal characters, we can also see that the agents are more diverse than in Indian tales. Especially family relations seem to play a more significant role. On the other hand, Indian tales show a strong preference towards male characters.

Since one character can belong to multiple classes, we can investigate the distribution of Proppian roles among other classes. Fig. 10 and Fig. 11 show the distribution of the hero class among other character classes. We can see the same preferences towards animal resp. male character classes as above.

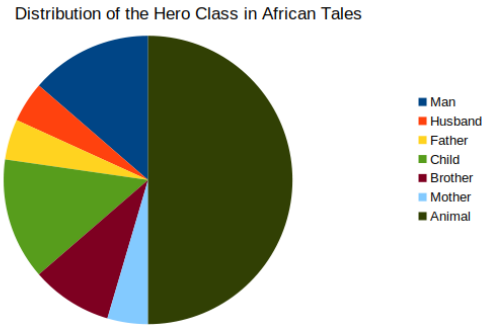


Figure 10. Distribution of the Hero class among other character classes in African Tales (multiple occurrences possible)

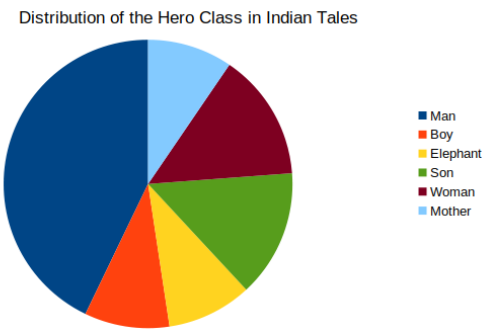


Figure 11. Distribution of the Hero class among other character classes in Indian Tales

Fig. 12 and 13 show the distribution of the villain class among other character classes. Interestingly, while the African tales follow the same pattern as before, i.e. the Villain mainly belonging to animal classes, the Indian tales show the same number of female and male villains.

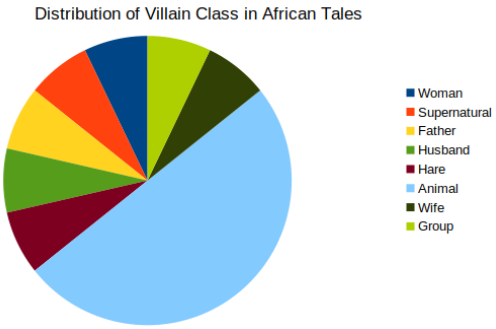


Figure 12. Distribution of the Villain class among other character classes in African Tales

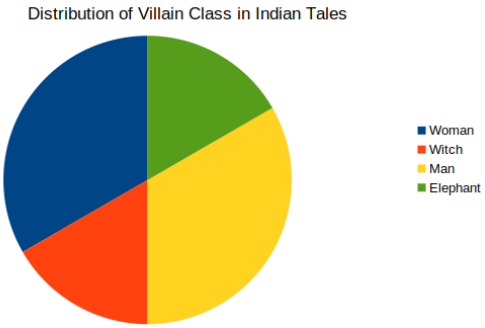


Figure 13. Distribution of the Villain class among other character classes in Indian Tales

8.1.2. Narrative Structure of Tales

We now want to investigate how the structure of tales differs throughout the small corpus. Propp divided the 31 functions into five categories, *Preparation*, *Complication*, *Functions of the Donor*, *Struggle*, and *Dénouement*. The annotated tales were analysed to determine how prevalent these five categories are. Fig. 14 shows the mean percentage of each of the categories among function sequences from African and Indian tales.

We can see that African tales focus more strongly on the preparatory functions, e.g. the description of the initial situation. Indian tales, on the other hand, stress the complicating functions more, e.g. the acts of villainy or the beginning counteraction. While 40 % of the mean function sequence length consists of preparative functions ($\alpha - \theta$), and 27 % consist of functions from the *Struggle* category, only 5 % of the sequence length is made up of function from the *Dénouement* category. In Table 3, only four of the eleven functions fall into that category. The rest of the final functions belong to the *Struggle* category, with the excep-

Table 2
Distribution of introductory and concluding functions in African tales

Beginning	End
Interdiction γ: 8	*Return* $\downarrow$: 4
Absentation β: 8	*Liquidation of Lack K*: 3
Trickery η: 4	*Transfiguration T*: 3
	Wedding W : 3
	Pursuit P: 1
	Provision of Magical Agent F: 1
	Unrecognized Arrival o: 1
	Rescue Rs: 1
	Punishment U: 1
	Victory I: 1
	Branding J: 1

tion of *Provision of Magical Agent F* (*Functions of the Donor*).

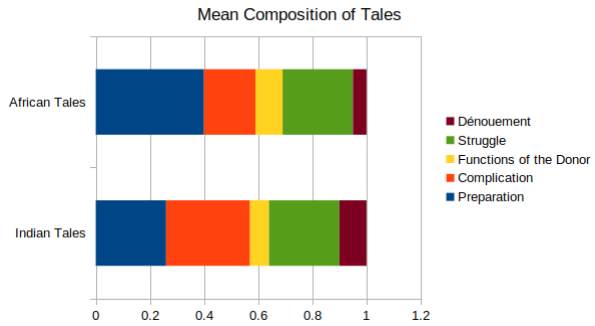


Figure 14. Composition of African and Indian tales by function classes

From 20 African tales that have been annotated so far, we can derive some structural features.

The tales' beginnings are mainly drawn from three functions: *Interdiction* γ, *Absentation* β, and *Trickery* η The only exception is Azuonye's analysis of the *Obaradeo* tale [9], where the initial function is *initial situation* α[11] and *Absentation* β is the second function. Also note that in Okodo's analysis [10] of this tale the initial function is indeed *Absentation* β.

For Indian tales, we find a slightly different division of initial and concluding functions, as shown in Table 3. The non-preparatory function *Villainy/Lack A* appears four times as an initial function, if we ignore the *Initial Situation* α, which appears 14 times in total. The other start functions fall into the *preparation* category.

[11] Propp himself states that the initial situation function is not technically a function.

Table 3
Distribution of introductory and concluding functions in Indian tales

Beginning	End
Absentation β: 9	*Return* $\downarrow$: 5
Villainy/Lack A: 4	*Transfiguration T*: 4
Reconnaissance ϵ: 1	*Solution N*: 2
Complicity θ: 1	*Punishment U*: 2
	Recognition Q : 1
	Liquidation of Lack K: 1

With regard to the concluding functions in Indian tales, we see that both *Return* $\downarrow$ and *Liquidation of Lack K* belong to the *Struggle* category. Only the remaining eight ending functions belong into the category of *Dénouement*.

8.1.3. Sequence Length

For African tales, the length of function sequences ranges between 16 and five functions. The average sequence length is 8.15 functions. Function sequences of Indian tales tend to be slightly longer, between eight and 18 functions with an average length of 10.2 functions. However, there are three two-move tales within the Indian corpus, which certainly increases the average sequence length.

8.1.4. Patterns of Functions

Spatial distance seems to play a certain role in all tales. The functions *Departure* $\uparrow$ and *Return* $\downarrow$ appear alone or together in eight of 20 African tales. In Indian tales, they appear 13 times, counting multi-move tales separately. In four African tales and nine Indian tales, both *Departure* $\uparrow$ and *Return* $\downarrow$ can be found as a pair.

The *Departure* $\uparrow$ function appears without a corresponding *Return* $\downarrow$ once in African tales and four times in Indian tales, while *Return* $\downarrow$ appears on its own three times in African tales.

Another prominent pattern is the pair *Villainy A/Villainy Lack a* and the corresponding function *Liquidation of Lack K* and their subfunctions. The pair appears together in nine African tales and ten Indian tales resp. moves. The distance between *Villainy A/Villainy Lack a* and *Liquidation of Lack K* ranges between one function and seven functions in African tales, and three to six functions in the Indian corpus.

Additionally, *Villainy A/Villainy Lack a* appears alone in five sequences of African tales and seven times in the Indian corpus. This indicates that in 25 % of the analysed African tales and 47 % of the Indian

tales, some form of harm is done to the hero or his/her family members without being resolved later.

In line with Propp's theory, there is no occurence of *Liquidation of Lack* without a preceding *Villainy A/Villainy Lack a* neither in African or Indian tales.

9. Limitations & Future Work

9.1. Fulltexts

By design, no fulltexts are stored in the context of the ontology. We do not see this as a limitation of the usability of the system. However, first time users might expect to be able to access the entire tale and not only the verbalisations stored when annotating functions and characters. This could be achieved by storing the fulltexts as annotated XML-TEI files and referencing the verbalisations by using pointers to the specific parts of the document. However, copyright aspects need to be taken into consideration when following this approach.

9.2. Natural Language Questions

Efforts have been made to generate the SPARQL queries answering the competency questions automatically. However, a natural language to SPARQL system would either have to rely on an extensive rule system or needs to be trained on a large set of questions and corresponding queries if a machine learning approach is used. Unfortunately, the implementation of this feature exceeds the scope of this project. However, for the system at hand such a feature would certainly be useful, especially since it would allow users with lower levels of IT-profiency to use it in a more intuitive manner. Attempts in this direction have been made by the ORAKEL project [31], or [32].

9.3. Future Work

As the ontology grows, potentially also linking additional media types such as video and voice recordings, one might consider taking into account additional features, such as features like facial expressions, reactions of the audience, interaction between narrator and audience, degree of attention, and composition of the audiencce "from the standpoint of age, sex, class or other social division" [8] should be added as datatype properties.

Furthermore, we want to add the possibility to visualize findings, e.g. by showing origins of tales on a map. Existing visualisation libraries for javascript could be used to create visually appealing graphics from the query results on browser side.

Measuring occurence of function pairs and their distance, as discussed in Section 4, could be automated with relatively low effort. This feature would certainly become more interesting as the ontology grows.

10. Conclusion

In this project, we aimed to show how ontologies can help formalise traditional theories from the Humanities. We chose to model Vladimir Propp's theory on the *Morphology of the Folktale* [3]. We demonstrated that we can easily access and compare data about folktales from different cultural backgrounds by translating traditional folkloristic questions about the structure of tales or the representation of characters into SPARQL queries. A carefully modelled ontology cannot only serve as means to access data and put it into context, but it can also assist traditional humanities scholars approaching research questions that are commonly solved by manual analysis and tedious comparison even today.

Our system allows users to compare different analyses of the same tale, and therefore holds potential to spark scientific discourse, providing a platform for different interpretations of Proppian functions, e.g. in the case of the *Obaraedo* tale as discussed in the beginning.

Proppian analyses are used both for teaching and research. Unfortunately, many of these analyses could previously not be contextualised and compared, because a digital tool to collect annotations was still missing.

We present an ontology that is accessible and invite folklorists to share their annotations on our webprotégé instance. This way, we also make our work accessible for folklorists who are interested in adding their own data and contextualising their analyses in an intercultural environment. Furthermore, we would like to invite folklorists and linguists to expand our set of translations for Proppian functions, Dramatis Personae, motifs and other concepts.

We believe that our ontology may be of interest for intercultural research on folktales, but also for translation studies because verbalisations of the same character or function can be provided together. While still

work in progress, our lightweight query system allows users to access the data and make their own deductions.

11. Acknowledgements

We would like to thank Mr Thierry Declerck, Prof. Jean Vincent Fonou Dombeu, and Mr Yasar Abbas for their help with the conceptionalization of the project. We are grateful to Ms Yuvika Singh and Ms Danielle Russel for help with the annotation of the African folktales. Special thanks to Mr Siyanda Sikobi for his help with the translation of the labels to isiZulu.

References

[1] J. Grimm and W. Grimm (eds), *Kinder- und Hausmärchen*, 7. edn, Verlag der Dieterichschen Buchhandlung, Göttingen, 1857.

[2] A.N. Afanasyev, *Russian Folk-Tales*, E. P. Dutton & Company, New York, 1916.

[3] V. Propp, *Morphology of the Folktale*, Vol. 10, University of Texas Press, 1968.

[4] S. Thompson, *Motif-Index of Folk-Literature: A Classification of Narrative Elements in Folk Tales, Ballads, Myths, Fables, Mediaeval Romances, Exempla, Fabliaux, Jest-Books, and Local Legends*, Vol. 1, Indiana University Press, 2001.

[5] H.-J. Uther, *The types of international folktales: a classification and bibliography, based on the system of Antti Aarne and Stith Thompson*, Suomalainen Tiedeakatemia, Academia Scientiarum Fennica, 2004.

[6] A. McCall Smith (ed.), *Children of Wax: African Folk Tales*, Canongate, 1989.

[7] F. Pannach, An Ontology-Driven Information System based on Vladimir Propp's Morphology of the Folktale for Southern African Folktales, Master's thesis, University of Göttingen, Germany, 2019. http://www.dbis.informatik.uni-goettingen.de/Teaching/Theses/PDF/MSc-Pannach-Folktales-mrz-2019.pdf.

[8] D.J. Crowley, The Uses of African Verbal Art, *Journal of the Folklore Institute* **6**(2/3) (1969), 118–132.

[9] C. Azuonye, Morphology of the Igbo Folktale: Ethnographic, Historiographic and Aesthetic Implications, *Folklore* **101**(1) (1990), 36–46.

[10] I. Okodo, Obaraedo: Conformity to Proppian Morphology, *AFRREV IJAH: An International Journal of Arts and Humanities* **1**(2) (2012), 100–111.

[11] A. Miles and S. Bechhofer, SKOS simple knowledge organization system reference, *W3C recommendation* **18** (2009), W3C.

[12] S. Weibel, The Dublin Core: a simple content description model for electronic resources, *Bulletin of the American Society for Information Science and Technology* **24**(1) (1997), 9–11.

[13] F. Peinado, P. Gervás and B. Díaz-Agudo, A description logic ontology for fairy tale generation, in: *Procs. of the Workshop on Language Resources for Linguistic Creativity, LREC*, Vol. 4, 2004, pp. 56–61.

[14] B. Dıaz-Agudo and P.A. González-Calero, CBROnto: a task/method ontology for CBR, *Proceedings of the 15th International Florida Artificial Intelligence Research Society Conference* **2** (2002), 101–106.

[15] T. Declerck, A. Aman, M. Banzer, D. Machácek, L. Schäfer and N. Skachkova, Multilingual Ontologies for the Representation and Processing of Folktales, in: *Proceedings of the First Workshop on Language technology for Digital Humanities in Central and (South-)Eastern Europe*, A. Dinu, P. Osenova and C. Vertan, eds, INCOMA Ltd, 2017, pp. 20–24.

[16] T. Declerck, A. Kostová and L. Schäfer, Towards a Linked Data Access to Folktales classified by Thompson's Motifs and Aarne-Thompson-Uther's Types, in: *Proceedings of Digital Humanities 2017*, ADHO, 2017.

[17] T. Declerck and L. Schäfer, Porting past Classification Schemes for Narratives to a Linked Data Framework, in: *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage*, ACM, 2017, pp. 123–127.

[18] N. Koleva, *Ontology-based iterative detection of characters and their recognition in folktales*, Bachelor Thesis, Saarland University, 2011.

[19] H. Scheub (ed.), *African Tales*, Univ. of Wisconsin Press, 2005.

[20] N. Greaves (ed.), *When the Hippo Was Hairy*, Lutterworth Press, 1990.

[21] P. Savory, *Bantu folk tales from southern Africa*, Howard Timmins, 1974.

[22] K. Sankunni, *Aithihyamaala*, Kottayam Nasanal Bukk Sttal, 1974.

[23] S. Ramachandran, *Aithihyamaala*, Mathrubhumi Books, 2014.

[24] E.G. Nair, *Poothappattu*, Poorna Publications, 1955.

[25] Story of the Poen "Poothappattu" (Ode to a "Pootham"). https://www.edasseri.org/English/poothapattu_story.htm.

[26] N. Noy and D.L. McGuinness, Ontology development 101: A Guide to Creating Your First Ontology, , KSL-01-05, Knowledge Systems Laboratory, Stanford University, 2001.

[27] M.A. Musen, The protégé project: a look back and a look forward, *AI Matters* **1**(4) (2015), 4–12. doi:10.1145/2757001.2757003.

[28] T. Tudorache, J. Vendetti and N.F. Noy, Web-Protege: A Lightweight OWL Ontology Editor for the Web., in: *OWLED*, 2008.

[29] D.C. Wimalasuriya and D. Dou, Ontology-based information extraction: An introduction and a survey of current approaches, *Journal of Information Science* **36**(3) (2010), 306–323. doi:10.1177/0165551509360123.

[30] A. Dundes, Folkloristics in the Twenty-First Century (AFS Invited Presidential Plenary Address, 2004), *The Journal of American Folklore* **118**(470) (2005), 385–408.

[31] P. Cimiano, P. Haase, J. Heizmann, M. Mantel and R. Studer, Towards portable natural language interfaces to knowledge bases–The case of the ORAKEL system, *Data & Knowledge Engineering* **65**(2) (2008), 325–354.

[32] J.-D. Kim and K.B. Cohen, Natural language query processing for SPARQL generation: A prototype system for SNOMED CT, in: *Proceedings of BioLINK 2013*, 2013, pp. 32–38.